



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Intellectual  
Property Office.

출원 번호 : 10-2003-0007415  
Application Number

출원 년 월 일 : 2003년 02월 06일  
Date of Application  
FEB 06, 2003

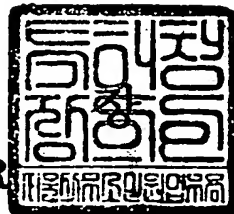
출원인 : 삼성전자주식회사  
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.



2003      년      05      월      30      일

특      허      청

COMMISSIONER



## 【서지사항】

|            |  |
|------------|--|
| 【서류명】      | 특허출원서  |
| 【권리구분】     | 특허   |
| 【수신처】      | 특허청장   |
| 【참조번호】     | 0022   |
| 【제출일자】     | 2003.02.06   |
| 【국제특허분류】   | G06F   |
| 【발명의 명칭】   | 제로 검출로 빠른 계산이 가능한 인크리멘터 및 그 방법   |
| 【발명의 영문명칭】 | Fast incrementer by zero detection and method thereof                            |
| 【출원인】      |  |
| 【명칭】       | 삼성전자 주식회사  |
| 【출원인코드】    | 1-1998-104271-3  |
| 【대리인】      |  |
| 【성명】       | 이영필  |
| 【대리인코드】    | 9-1998-000334-6  |
| 【포괄위임등록번호】 | 2003-003435-0  |
| 【대리인】      |  |
| 【성명】       | 정상빈  |
| 【대리인코드】    | 9-1998-000541-1  |
| 【포괄위임등록번호】 | 2003-003437-4  |
| 【발명자】      |  |
| 【성명의 국문표기】 | 권요한  |
| 【성명의 영문표기】 | KWON, Yo Han   |
| 【주민등록번호】   | 731211-1069413   |
| 【우편번호】     | 435-050  |
| 【주소】       | 경기도 군포시 금정동 삼익소월@ 372-1002   |
| 【국적】       | KR   |
| 【심사청구】     | 청구   |
| 【취지】       | 특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사를 청구합니다. 대리인<br>이영필 (인) 대리인<br>정상빈 (인) |

## 【수수료】

|         |    |   |        |   |
|---------|----|---|--------|---|
| 【기본출원료】 | 20 | 면 | 29,000 | 원 |
|---------|----|---|--------|---|

|         |    |   |        |   |
|---------|----|---|--------|---|
| 【가산출원료】 | 11 | 면 | 11,000 | 원 |
|---------|----|---|--------|---|

|          |   |   |   |   |
|----------|---|---|---|---|
| 【우선권주장료】 | 0 | 건 | 0 | 원 |
|----------|---|---|---|---|

|         |    |   |         |   |
|---------|----|---|---------|---|
| 【심사청구료】 | 14 | 항 | 557,000 | 원 |
|---------|----|---|---------|---|

|      |         |   |  |  |
|------|---------|---|--|--|
| 【합계】 | 597,000 | 원 |  |  |
|------|---------|---|--|--|

|        |                   |  |  |  |
|--------|-------------------|--|--|--|
| 【첨부서류】 | 1. 요약서·명세서(도면)_1통 |  |  |  |
|--------|-------------------|--|--|--|

**【요약서】****【요약】**

제로 검출로 빠른 계산이 가능한 인크리멘터 및 그 방법이 개시된다. 상기 인크리멘터는 오퍼랜드, 4비트 그룹별 제1 논리 상태 존재 정보, 4비트 그룹별 플래그 정보, 및 소정의 인크리멘트 값에 대하여 소정 논리식에 따른 소정의 논리 조합을 수행하여 4비트 그룹별로, 제1 논리 상태인 4비트, 오퍼랜드 그대로의 4비트, 또는 소정의 인크리멘트 값의 4비트를 출력하여 전체 인크리멘트 값을 발생시킨다. 따라서, 제로 검출하는 간단한 로직과 맥스 구조의 간단한 회로에 의하여 다이내믹하게 계산하므로, 빠른 계산이 가능하고, 칩 상의 면적을 줄일 수 있는 효과가 있다.

**【대표도】**

도 1

**【명세서】****【발명의 명칭】**

제로 검출로 빠른 계산이 가능한 인크리멘터 및 그 방법{Fast incrementer by zero detection and method thereof}

**【도면의 간단한 설명】**

본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 간단한 설명이 제공된다.

도 1은 본 발명의 일실시예에 따른 인크리멘터(incrementer)의 블록도이다.

도 2는 도 1의 4비트 제로(zero) 검출부의 구체적인 회로도이다.

도 3a 및 도 3b는 도 1의 플래그(flag) 정보 생성부의 구체적인 회로도이다.

도 4는 도 1의 4비트 인크리먼트부의 구체적인 회로도이다.

도 5는 도 1의 인크리먼트 출력부의 구체적인 회로도이다.

**【발명의 상세한 설명】****【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

<7> 본 발명은 마이크로 프로세서(microprocessor)의 연산기에 관한 것으로, 특히 인크리멘터(incrementer)에 관한 것이다.

<8> 인크리멘터(incrementer)는 입력되는 오퍼랜드(operand)에 "1"을 더하는 계산을 하는 애더(adder) 또는 카운터(counter)이다. 마이크로 프로세서(micorprocessor) 등에서 인크리멘터(incrementer)는 불린 로직(boolean logic)의 투스 컴플리먼트(2's

complement)를 계산하거나, 기타 오퍼랜드(operand)에 1을 더하는 계산을 포함하는 다양한 연산을 수행하는데 사용된다.

<9> 풀 애더(full adder)를 사용하는 종래의 인크리멘터(incrementer)는 캐리 비트의 전달을 기다리는 시간의 소모로 인하여 동작 스피드가 떨어지고, 캐리 비트를 처리하기 위한 회로 등을 구비함에 따라 그 전체 회로가 칩(chip) 상에서 상당한 면적을 차지하는 문제점이 있다.

<10> 또한, 미국 특허, "US5,635,858A"에 잘 나타나 있는 바와 같은, 종래의 제로 스톱핑(zero stopping) 인크리멘터(incrementer)는, 입력되는 오퍼랜드(operand)가 짝수인지 홀수인지를 구별하여 짝수인 경우에는 LSB(least significant bit)를 "1"로 되게 하고, 홀수인 경우에는 최초 "0"이 검출될 때 그 "0"을 "1"로 되게 한 다음, 그 앞쪽 LSB(least significant bit)들의 "0"들을 모두 "1"로 되게 하는 방법을 취한다. 그러나, 종래의 제로 스톱핑(zero stopping) 인크리멘터(incrementer)는, 많은 양의 스테틱(static)한 로직 게이트(gate) 사용으로 칩(chip) 상에서 상당한 면적을 차지하는 문제점이 있다.

#### 【발명이 이루고자 하는 기술적 과제】

<11> 따라서, 본 발명이 이루고자하는 기술적 과제는, 제로(zero) 검출하는 간단한 로직과 믹스 구조의 간단한 회로에 의하여 다이내믹하게 빠른 계산이 가능하고, 칩(chip) 상의 면적을 줄일 수 있는 인크리멘터(incrementer)를 제공하는 데 있다.

- <12> 본 발명이 이루고자하는 다른 기술적 과제는, 제로(zero) 검출하는 간단한 로직과  
믹스 구조의 간단한 회로에 의하여 다이내믹하게 빠른 계산이 가능하고, 칩(chip) 상의  
면적을 줄일 수 있는 인크리멘터(incrementer)의 인크리먼트 방법을 제공하는 데 있다.

**【발명의 구성 및 작용】**

- <13> 상기의 기술적 과제를 달성하기 위한 본 발명에 따른 인크리멘터(incrementer)는,  
4비트 제로(zero) 검출부, 플래그(flag) 정보 생성부, 4비트 인크리먼트부, 및 인크리멘  
트 출력부를 구비한다.
- <14> 상기 4비트 제로(zero) 검출부는 입력되는 오퍼랜드(operand)(IN)를 하위부터 4비  
트씩 그룹들을 지어 그룹들 각각에 제1 논리 상태가 존재하는지를 판단하여, 상기 제1  
논리 상태가 존재하면 제2 논리 상태로 발생시키고, 상기 제1 논리 상태가 존재하지 하  
지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보(ZD)를  
출력한다.
- <15> 상기 플래그(flag) 정보 생성부는 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)  
의 하위에서 최초로 나타나는 상기 제2 논리 상태에 대응하여, 상기 제2 논리 상태가 속  
한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그  
룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그  
(flag) 정보(CA)를 출력한다.
- <16> 상기 4비트 인크리먼트부는 입력되는 상기 오퍼랜드(operand)(IN)를 받아 상기 4비  
트 그룹별 소정의 인크리먼트를 수행한다.

<17>      상기 인크리먼트 출력부는 상기 오퍼랜드(operand)(IN), 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD), 상기 4비트 그룹별 플래그(flag) 정보(CA), 및 상기 소정의 인크리먼트 값(ADD)을 소정의 논리 조합하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트, 상기 오퍼랜드(operand)(IN) 그대로의 4비트, 또는 상기 소정의 인크리먼트 값(ADD)의 4비트를 출력하여 전체 인크리먼트 값(IO)을 발생시킨다.

<18>      여기서, 상기 소정의 인크리먼트는, 논리식,

<19>      IF IN<0> and IN<1> = "1",

<20>      (IN+1)<3:2> = INC<3:2> ,

<21>      (IN+1)<1:0> = "00"

<22>      IF IN<0> and IN<1> = "0",

<23>      (IN+1)<3:2> = IN<3:2> ,

<24>      (IN+1)<1:0> = INC<1:0>

<25>      (여기서 IN은 오퍼랜드, "IN+1"은 인크리먼트 값, INC는 새로 정의된 인크리먼트, "0"은 제1 논리 상태, "1"은 제2 논리 상태)

<26>      및, 논리식,

<27>      INC<0> = ~IN<0>

<28>      INC<1> = IN<0> OR IN<1>

<29>      INC<2> = ~IN<2>

<30>      INC<3> = IN<2> OR IN<3>

<31>      (여기서 IN은 오퍼랜드, INC는 새로 정의된 인크리먼트)



- <32>      에 의하여 수행되는 것을 특징으로 한다.
- <33>      상기 소정의 논리 조합은, 논리식,
- <34>      IF ZD = "0" and CA = "0", IO = "0000",
- <35>      IF ZD = "1" and CA = "0", IO = IN+1,
- <36>      IF CA = "1"(ZD = don't care), IO = IN
- <37>      (여기서, ZD는 4비트 그룹별 제1 논리 상태 존재 정보, CA는 4비트 그룹별 플래그 정보, IO는 전체 인크리먼트 값의 4비트 그룹별 출력, IN은 오퍼랜드, "IN+1"은 인크리먼트 값, "0"은 제1 논리 상태, "1"은 제2 논리 상태)
- <38>      에 의하여 수행되는 것을 특징으로 한다.
- <39>      상기 인크리멘터(incrementer)는, 상기 4비트 제로(zero) 검출부, 상기 플래그(flag) 정보 생성부, 상기 4비트 인크리먼트부, 및 상기 인크리먼트 출력부가 소정의 클럭에 의하여 액티브 될 때, 동작하는 것을 특징으로 한다. 상기 소정의 클럭이 비활성화 상태일 때, 상기 4비트 제로(zero) 검출부, 상기 플래그(flag) 정보 생성부, 상기 4비트 인크리먼트부, 및 상기 인크리먼트 출력부 각각의 출력 버퍼들 앞단에서 소정의 전압으로 프리차징된다. 상기 출력 버퍼들 각각은, 상기 프리차징 전압을 인버팅하는 인버터와 상기 인버터 출력의 제어를 받아 상기 인버터의 입력단에 상기 소정의 전압을 공급하는 PMOSFET를 구비한다. 상기 인버터 출력은, 상기 소정의 클럭이 활성화 상태이고, 상기 출력 버퍼들 각각의 앞단과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET들 모두가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 한다.

- <40>      상기의 다른 기술적 과제를 달성하기 위한 본 발명에 따른 인크리멘터(incrementer)의 인크리먼트 방법은, 다음과 같은 단계를 구비한다.
- <41>      즉, 본 발명에 따른 인크리멘터(incrementer)의 인크리먼트 방법은, 먼저, 인크리멘터(incrementer)가 입력되는 오퍼랜드(operand)(IN)를 하위부터 4비트씩 그룹들을 지어 그룹들 각각에 제1 논리 상태가 존재하는지를 판단하여, 상기 제1 논리 상태가 존재하면 제2 논리 상태로 발생시키고, 상기 제1 논리 상태가 존재하지 하지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보(ZD)를 출력한다.
- <42>      다음에, 상기 인크리멘터(incrementer)는, 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)의 상기 제2 논리 상태에 대응하여 상기 제2 논리 상태가 속한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그(flag) 정보(CA)를 출력한다.
- <43>      이에 따라, 상기 인크리멘터(incrementer)는, 입력되는 상기 오퍼랜드(operand)(IN)를 받아 상기 4비트 그룹별 소정의 인크리먼트를 수행한다. 또한, 상기 오퍼랜드(operand)(IN), 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD), 상기 4비트 그룹별 플래그(flag) 정보(CA), 및 상기 소정의 인크리먼트 값(ADD)을 소정의 논리 조합하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트, 상기 오퍼랜드(operand)(IN) 그대로의 4비트, 또는 상기 소정의 인크리먼트 값(ADD)의 4비트를 출력하여 전체 인크리먼트 값(IO)을 발생시킨다.

- <44> 본 발명과 본 발명의 동작상의 이점 및 본 발명의 실시에 의하여 달성되는 목적을 충분히 이해하기 위해서는 본 발명의 바람직한 실시예를 예시하는 첨부 도면 및 첨부 도면에 기재된 내용을 참조하여야만 한다.
- <45> 이하, 첨부한 도면을 참조하여 본 발명의 바람직한 실시예를 설명함으로써, 본 발명을 상세히 설명한다. 각 도면에 제시된 동일한 참조부호는 동일한 부재를 나타낸다.
- <46> 도 1은 본 발명의 일실시예에 따른 인크리멘터(incrementer)의 블록도이다.
- <47> 도 1을 참조하면, 본 발명의 일실시예에 따른 인크리멘터(incrementer)는, 4비트 제로(zero) 검출부(110), 플래그(flag) 정보 생성부(120), 4비트 인크리먼트부(130), 및 인크리먼트 출력부(140)를 구비한다.
- <48> 상기 4비트 제로(zero) 검출부(110)는 입력되는 오퍼랜드(operand)(IN)를 하위부터 4비트씩 그룹들을 지어 그룹들 각각에 제로(zero), 즉, 제1 논리 상태(논리 로우 상태)가 존재하는지를 판단하여, 상기 제1 논리 상태가 존재하면 "1", 즉, 제2 논리 상태(논리 하이 상태)로 발생시키고, 상기 제1 논리 상태가 존재하지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보(ZD)를 출력한다.
- <49> 즉, 상기 오퍼랜드(operand)(IN)는 마이크로 프로세서(micorprocessor)의 연산에서 32 비트 혹은 64 비트 값을 가지며, 이외에도 본 발명의 일실시예에 따른 인크리멘터(incrementer)의 연산은 상기 오퍼랜드(operand)(IN)의 비트 수에 제한되지 않으나, 여기서는 상기 오퍼랜드(operand)(IN)가 32 비트의 데이터로 입력되는 것으로 가정한다.
- <50> 따라서, 상기 4비트 제로(zero) 검출부(110)에서 출력되는 4비트 그룹별 제1 논리 상태 존재 정보(ZD)는, 입력되는 32 비트의 오퍼랜드(operand)(IN)를 4 비트씩 8그룹으로

로 나눌 때의 8비트 정보로서, 제로(zero)가 존재하는 그룹에 대하여 "1"로 되고 제로(zero)가 존재하지 않는 그룹에 대하여 "0"으로 되는 값을 가지는 8 비트의 정보이다.

<51>      아래 [예제 1]에서, 오퍼랜드(operand)(IN)의 하위부터 4비트씩 그룹을 지어 판단할 때, ZD<6> ZD<3> 각각이 속하는 그룹에서 "0"이 나타나므로, ZD<6> ZD<3>은 "1"로 되고, 나머지 ZD<7> ZD<5> ZD<4> ZD<2> ZD<1> ZD<0>은 "0"으로 된다.

<52> [예제 1]

<53>      오퍼랜드 :            1111 0000 1111 1111 1011 1111 1111 1111

<54>      전체 인크리먼트 값 : 1111 0000 1111 1111 1100 0000 0000 0000

<55>      CA<7:1> :            CA<7> CA<6> CA<5> CA<4> CA<3> CA<2> CA<1>

<56>      ZD<7:0> :            ZD<7> ZD<6> ZD<5> ZD<4> ZD<3> ZD<2> ZD<1> ZD<0>

<57>      상기 플래그(flag) 정보 생성부(120)는 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)의 하위에서 최초로 나타나는 상기 제2 논리 상태에 대응하여, 상기 제2 논리 상태가 속한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그(flag) 정보(CA)를 출력한다.

<58>      즉, 상기 플래그(flag) 정보 생성부(120)에서 출력되는 4비트 그룹별 플래그(flag) 정보(CA)는, 입력되는 32 비트의 오퍼랜드(operand)(IN)를 4 비트씩 8그룹으로 나눌 때의 8비트 정보로서, 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)에 의하여 상기 ZD가 최초로 "1"로 되는 그룹의 상위 그룹(ZD<4>가 속하는 그룹)부터 전체 상위 그룹까지

에 대하여 "1"로 되고, 상기 ZD가 최초로 "1"로 되는 그룹(ZD<3>이 속하는 그룹)을 포함하여 전체 하위 그룹까지에 대하여 "0"으로 되는 8 비트의 정보이다.

<59> 위의 [예제 1]에서, 오퍼랜드(operand)(IN)의 하위부터 4비트씩 그룹을 지어 판단하여 ZD<6> ZD<3> 각각이 속하는 그룹에서 "0"이 나타나지만, 그 하위는 ZD<3>이므로, ZD<3>이 속하는 그룹을 포함하여 전체 하위 그룹까지, 즉, CA<3> CA<2> CA<1>은 "0"으로 되고, 그 상위 그룹부터 전체 상위 그룹까지, 즉, CA<7> CA<6> CA<5> CA<4>은 "1"로 된다.

<60> 상기 4비트 인크리먼트부(130)는 입력되는 상기 오퍼랜드(operand)(IN)를 받아 상기 4비트 그룹별 소정의 인크리먼트를 수행한다. 여기서 수행되어 출력되는 소정의 인크리먼트 값(ADD)은 상기 인크리먼트 출력부(140)로 입력된다.

<61> 여기서, 상기 소정의 인크리먼트는, [논리식 1], 및 [논리식 2]에 의하여 수행되는 것을 특징으로 한다.

<62> [논리식 1]

<63> IF IN<0> and IN<1> = "1",

<64> (IN+1)<3:2> = INC<3:2>,

<65> (IN+1)<1:0> = "00"

<66> IF IN<0> and IN<1> = "0",

<67> (IN+1)<3:2> = IN<3:2>,

<68> (IN+1)<1:0> = INC<1:0>

<69> (여기서 IN은 오퍼랜드, "IN+1"은 인크리먼트 값, INC는 새로 정의된 인크리먼트, "0"은 제1 논리 상태, "1"은 제2 논리 상태)

<70> [논리식 2]

<71>  $INC<0> = \sim IN<0>$

<72>  $INC<1> = IN<0> \text{ OR } IN<1>$

<73>  $INC<2> = \sim IN<2>$

<74>  $INC<3> = IN<2> \text{ OR } IN<3>$

<75> (여기서 IN은 오퍼랜드, INC는 새로 정의된 인크리먼트)

<76> 상기 인크리먼트 출력부(140)는 상기 오퍼랜드(operand)(IN), 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD), 상기 4비트 그룹별 플래그(flag) 정보(CA), 및 상기 소정의 인크리먼트 값(ADD)을 소정의 논리 조합하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트, 상기 오퍼랜드(operand)(IN) 그대로의 4비트, 또는 상기 소정의 인크리먼트 값(ADD)의 4비트를 출력하여 전체 인크리먼트 값(IO)을 발생시킨다.

<77> 상기 소정의 논리 조합은, [논리식 3]에 의하여 수행되는 것을 특징으로 한다.

<78> [논리식 3]

<79> IF ZD = "0" and CA = "0", IO = "0000",

<80> IF ZD = "1" and CA = "0", IO = IN+1,

<81> IF CA = "1"(ZD = don't care), IO = IN

- <82> (여기서, ZD는 4비트 그룹별 제1 논리 상태 존재 정보, CA는 4비트 그룹별 플래그 정보, IO는 전체 인크리먼트 값의 4비트 그룹별 출력, IN은 오퍼랜드, "IN+1"은 인크리먼트 값, "0"은 제1 논리 상태, "1"은 제2 논리 상태)
- <83> 위의 [예제 1]에서, ZD<3>이 속하는 그룹에서 오퍼랜드(operand)(IN)에 "1"을 더하여 전체 인크리먼트 값(IO)이 "1111 0000 1111 1111 1100 0000 0000 0000"과 같이 계산됨을 알 수 있다. 이와 같이 계산하는 본 발명의 일실시예에 따른 인크리멘터(incrementer)의 동작 스피드는 200ps(pico sec)이하로 나타났고, 이때 전체 시스템의 동작 주파수는 10GHz 이상으로 나타날 것으로 기대된다.
- <84> 이와 같은 본 발명의 일실시예에 따른 인크리멘터(incrementer)의 동작을 도 2 내지 도 5에 도시된 구체적인 회로도에 의하여 좀더 자세히 설명한다.
- <85> 도 2는 도 1의 4비트 제로(zero) 검출부(110)의 구체적인 회로도이다.
- <86> 도 2를 참조하면, 도 1의 4비트 제로(zero) 검출부(110)는, 소정의 클럭(CLK)이 비활성화 상태, 즉, 제1 논리 상태일 때, 출력 버퍼(INV21,P2) 앞단이 소정의 회로(P1)에 의하여 소정의 전압(VDD)으로 프리차징된다. 출력 버퍼(INV21,P2)는, 상기 프리차징 전압(VDD)을 인버팅하는 인버터(INV21)와 상기 인버터(INV21) 출력의 제어를 받아 상기 인버터(INV21)의 입력단에 상기 소정의 전압(VDD)을 공급하는 PMOSFET(P2)를 구비한다. 상기 인버터(INV21) 출력은, 상기 소정의 클럭(CLK)이 활성화 상태, 즉, 제2 논리 상태이고, 상기 출력 버퍼(INV21,P2)의 앞단과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET들(N1 및 N2 등) 모두가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 한다. NIN<0> 내지 NIN<3>은 상기 오퍼랜드(operand)(IN) 32 비트 중 임의의 그

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants: YO-HAN KWON  
Application No.: NEW APPLICATION  
Filed: October 30, 2003  
For: FAST INCREMENTER USING ZERO DETECTION AND  
INCRÉMENT METHOD THEREOF

**PRIORITY LETTER**

October 30, 2003

Honorable Commissioner of Patents and Trademarks  
Washington, DC 20231

Dear Sirs:

Pursuant to the provisions of 35 U.S.C. 119, enclosed is/are a certified copy of the following priority document(s).

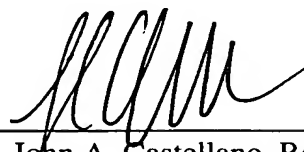
| <u>Application No.</u> | <u>Date Filed</u> | <u>Country</u>    |
|------------------------|-------------------|-------------------|
| 10-2003-0007415        | February 6, 2003  | REPUBLIC OF KOREA |

In support of Applicant's priority claim, please enter this document into the file.

Respectfully submitted,

HARNESS, DICKEY, & PIERCE, P.L.C.

By



John A. Castellano, Reg. 35,094  
P.O. Box 8910  
Reston, Virginia 20195  
(703) 390-3030

JAC/jj



룹에 속하는 4비트의 인버팅 신호들이다. 여기서, 4비트에 대한 회로에 대하여 예를 들어 도시하였으나, 32 비트 오퍼랜드(operand)(IN)에 대한 8개의 그룹들에 대하여 각각 도 2와 같은 회로가 필요하다.

<87> 즉, 도 2와 같은 4비트 제로(zero) 검출부(110)는, 입력되는 오퍼랜드(operand)(IN)를 하위부터 4비트씩 그룹들을 지어 그룹들 각각에 제로(zero), 즉, 제1 논리 상태(논리 로우 상태)가 존재하는지를 판단하여, 상기 제1 논리 상태가 존재하면 "1", 즉, 제2 논리 상태(논리 하이 상태)로 발생시키고, 상기 제1 논리 상태가 존재하지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보(ZD)를 출력할 수 있다.

<88> 도 3a 및 도 3b는 도 1의 플래그(flag) 정보 생성부(120)의 구체적인 회로도이다.

<89> 도 3a 및 도 3b를 참조하면, 도 1의 플래그(flag) 정보 생성부(120)는, 클럭 버퍼(310)를 통하여 입력되는 소정의 클럭(CLK)이 비활성화 상태, 즉, 제1 논리 상태일 때, 출력 버퍼(INV31,P32) 앞단(ZZCA 노드)이 소정의 회로(320)에 의하여 소정의 전압(VDD)으로 프리차징된다. 출력 버퍼(INV31,P32)는, 상기 프리차징 전압(VDD)을 인버팅하는 인버터(INV31)와 상기 인버터(INV31) 출력의 제어를 받아 상기 인버터(INV31)의 입력단(ZZCA 노드)에 상기 소정의 전압(VDD)을 공급하는 PMOSFET(P32)를 구비한다. 상기 인버터(INV31) 출력은, 상기 소정의 클럭(CLK)이

활성화 상태, 즉, 제2 논리 상태이고, 상기 출력 버퍼(INV31,P32)의 앞단(ZZCA 노드)과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET(330)가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 한다. ZD<0> 내지 ZD<6>은 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)이다. 여기서, 최상위 4비트에서 "0"이 최초로 나타나는 경우는 플래그(flag) 정보 CA<7>이 "0"이 될 것이므로, 아래에서 기술 할 도 5와 같이 동작하는 회로에서, ZD<7>은 4비트 그룹별 플래그(flag) 정보(CA) 생성에 불필요하고, 또한, CA<0>는 항상 "0"이므로 CA<0> 신호는 불필요하다.

<90> 즉, 도 3a 및 도 3b와 같은 플래그(flag) 정보 생성부(120)는 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)의 하위에서 최초로 나타나는 상기 제2 논리 상태에 대응하여, 상기 제2 논리 상태가 속한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그(flag) 정보(CA)를 출력할 수 있다.

<91> 도 4a 및 도 4b는 도 1의 4비트 인크리먼트부(130)의 구체적인 회로도이다.

<92> 도 4a 및 도 4b를 참조하면, 도 1의 4비트 인크리먼트부(130)는, 비트별 인크리먼트 회로(420~450)를 구비하고, 클럭 버퍼(410)를 통하여 입력되는 소정의 클럭(CLK)이 비활성화 상태, 즉, 제1 논리 상태일 때, 출력 버퍼들(INV41,P12 등) 앞단이 소정의 회로(P11 등)에 의하여 소정의 전압(VDD)으로 프리차징된다. 출력 버퍼들(INV41,P12 등)은, 상기 프리차징 전압(VDD)을 인버팅하는 인버터(INV41 등)와 상기 인버터(INV41 등) 출력의 제어를 받아 상기 인버터(INV41 등)의 입력단에 상

기 소정의 전압(VDD)을 공급하는 PMOSFET(P12 등)를 구비한다. 상기 인버터(INV41 등) 출력은, 상기 소정의 클럭(CLK)이 활성화 상태, 즉, 제2 논리 상태이고, 상기 출력 버퍼들(INV41, P12 등)의 앞단과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET들(N111 및 N12 등) 모두가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 한다. IN<0> 내지 IN<3> 및 NIN<0> 과 NIN<2> 각각은 상기 오퍼랜드(operand)(IN) 32 비트 중 임의의 그룹에 속하는 4비트 신호들 및 그 인버팅 신호들이다. 여기서, 4비트에 대한 회로에 대하여 예를 들어 도시하였으나, 32 비트 오퍼랜드(operand)(IN)에 대한 8개의 그룹들에 대하여 각각 도 4a 및 도 4b와 같은 회로가 필요하다.

<93> 즉, 도 4a 및 도 4b와 같은 4비트 인크리먼트부(130)는, 입력되는 상기 오퍼랜드(operand)(IN)를 받아 상기 4비트 그룹별로 [논리식 1], 및 [논리식 2]에 의하여 수행되는 인크리먼트를 수행한다. [논리식 1]은 4비트 중 하위 2비트에 의하여 "0"이 있는지를 판단하여, "0"이 있는 경우와 "0"이 없는 경우에 대하여 각 비트에 대한 인크리먼트 값을 정의한다. [논리식 2]는 각 비트에 대한 인크리먼트 값을 구하는 방법을 정의한다. [논리식 2]에서 "~"는 인버팅 값을 의미한다.

<94> 예를 들어, 4비트 중 하위 2비트에 "0"이 있으면, 상기 4비트 인크리먼트부(130)는 상위 2비트 출력으로 오퍼랜드(operand)(IN)을 그대로 출력하고, 하위 2비트 출력으로 [논리식 2]에 정의된 인크리먼트 값을 출력한다. 4비트 중 하위 2비트에 "0"이 없으면, 상기 4비트 인크리먼트부(130)는 상위 2비트 출력으로 [논리식 2]에 정의된 인크리먼트 값을 출력하고, 하위 2비트 출력으로 "00"을 출력한다.

<95> 도 5는 도 1의 인크리먼트 출력부(140)의 구체적인 회로도이다.

<96> 도 5를 참조하면, 도 1의 인크리먼트 출력부(140)는, 클럭 버퍼(510)를 통하여 입력되는 소정의 클럭(CLK)이 비활성화 상태, 즉, 제1 논리 상태일 때, 출력 버퍼(INV21,P2) 앞단이 소정의 회로(P51)에 의하여 소정의 전압(VDD)으로 프리차징된다. 출력 버퍼(INV51,P52)는, 상기 프리차징 전압(VDD)을 인버팅하는 인버터(INV51)와 상기 인버터(INV51) 출력의 제어를 받아 상기 인버터(INV51)의 입력단에 상기 소정의 전압(VDD)을 공급하는 PMOSFET(P52)를 구비한다. 상기 인버터(INV51) 출력은, 상기 소정의 클럭(CLK)이 활성화 상태, 즉, 제2 논리 상태이고, 상기 출력 버퍼(INV51,P52)의 앞단과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET들(530 또는 540) 모두가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 한다. NCA는 상기 CA의 인버팅 신호이다. 여기서, 1비트에 대한 회로에 대하여 예를 들어 도시하였으나, 32 비트 오퍼랜드(operand)(IN)를 받는 32 비트 각각에 대하여 도 5와 같은 회로가 필요하다. 이때, NCA, CA, 및 ZD 각각은 4비트 그룹들에 대한 정보로서 8비트씩이고, IN 및 ADD 각각은 같은 비트별로 대응되어 있으며 32 비트씩이다.

<97> 즉, 도 5와 같은 인크리먼트 출력부(140)는 상기 오퍼랜드(operand)(IN), 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD), 상기 4비트 그룹별 플래그(flag) 정보(CA), 및 상기 소정의 인크리먼트 값(ADD)을 소정의 논리 조합하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트([논리식 3]에서 "0000"), 상기 오퍼랜드(operand)(IN) 그대로의 4비트([논리식 3]에서 "IN"), 또는 상기 소정의 인크리먼트 값(ADD)의 4비트([논리식 3]에서 "IN+1")를 출력하여 전체 인크리먼트 값(I0)을 발생시킬 수 있다.

<98> 위에서 기술한 바와 같이, 본 발명의 일실시예에 따른 인크리멘터(incrementer)는, 먼저 4비트 제로(zero) 검출부(110)가 입력되는 오퍼랜드(operand)(IN)를 하위부터 4비

트씩 그룹들을 지어 그룹들 각각에 제1 논리 상태가 존재하는지를 판단하여, 상기 제1 논리 상태가 존재하면 제2 논리 상태로 발생시키고, 상기 제1 논리 상태가 존재하지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보(ZD)를 출력한다. 플래그(flag) 정보 생성부(120)는 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)의 하위에서 최초로 나타나는 상기 제2 논리 상태에 대응하여, 상기 제2 논리 상태가 속한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그(flag) 정보(CA)를 출력한다. 4비트 인크리먼트부(130)는 입력되는 상기 오퍼랜드(operand)(IN)를 받아 [논리식 1] 및 [논리식 2]에 따라 상기 4비트 그룹별 소정의 인크리먼트를 수행한다. 이에 따라, 인크리먼트 출력부(140)는 상기 오퍼랜드(operand)(IN), 상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD), 상기 4비트 그룹별 플래그(flag) 정보(CA), 및 상기 소정의 인크리먼트 값(ADD)에 대하여 [논리식 3]에 따른 소정의 논리 조합을 수행하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트, 상기 오퍼랜드(operand)(IN) 그대로의 4비트, 또는 상기 소정의 인크리먼트 값(ADD)의 4비트를 출력하여 전체 인크리먼트 값(IO)을 발생시킨다.

<99> 이상에서와 같이 도면과 명세서에서 최적 실시예가 개시되었다. 여기서 특정한 용어들이 사용되었으나, 이는 단지 본 발명을 설명하기 위한 목적에서 사용된 것이지 의미 한정이나 특허청구범위에 기재된 본 발명의 범위를 제한하기 위하여 사용된 것은 아니다. 그러므로 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

**【발명의 효과】**

<100> 상술한 바와 같이 본 발명에 따른 인크리멘터(incrementer)는, 제로(zero) 검출하는 간단한 로직과 먹스 구조의 간단한 회로에 의하여 다이내믹하게 계산하므로, 빠른 계산이 가능하고, 칩(chip) 상의 면적을 줄일 수 있는 효과가 있다.

**【특허청구범위】****【청구항 1】**

입력되는 오퍼랜드를 하위부터 4비트씩 그룹들을 지어 그룹들 각각에 제1 논리 상태가 존재하는지를 판단하여, 상기 제1 논리 상태가 존재하면 제2 논리 상태로 발생시키고, 상기 제1 논리 상태가 존재하지 하지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보(ZD)를 출력하는 4비트 제로 검출부;

상기 4비트 그룹별 제1 논리 상태 존재 정보(ZD)의 하위에서 최초로 나타나는 상기 제2 논리 상태에 대응하여, 상기 제2 논리 상태가 속한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그 정보를 출력하는 플래그 정보 생성부;

입력되는 상기 오퍼랜드를 받아 상기 4비트 그룹별 소정의 인크리먼트를 수행하는 4비트 인크리먼트부; 및

상기 오퍼랜드, 상기 4비트 그룹별 제1 논리 상태 존재 정보, 상기 4비트 그룹별 플래그 정보, 및 상기 소정의 인크리먼트 값을 소정의 논리 조합하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트, 상기 오퍼랜드 그대로의 4비트, 또는 상기 소정의 인크리먼트 값의 4비트를 출력하여 전체 인크리먼트 값을 발생시키는 인크리먼트 출력부를 구비하는 것을 특징으로 하는 인크리멘터.

**【청구항 2】**

제 1항에 있어서, 상기 소정의 인크리먼트는,

논리식 ,

IF IN<0> and IN<1> = "1",

(IN+1) <3:2> = INC<3:2>,

(IN+1)<1:0> = "00"

IF IN <0> and IN<1> = "0",

(IN+1)<3:2> = IN<3:2> ,

(IN+1) <1:0> = INC<1:0>

(여기서 IN은 오퍼랜드, "IN+1"은 인크리먼트 값, INC는 새로 정의된 인크리먼트,  
"0"은 제1 논리 상태, "1"은 제2 논리 상태)

및 , 논리식,

INC<0> = ~IN<0>

INC <1> = IN<0> OR IN<1>

INC<2> = ~IN<2>

INC <3> = IN<2> OR IN<3>

(여기서 IN은 오퍼랜드, INC는 새로 정의된 인크리먼트)

에 의하여 수행되는 것을 특징으로 하는 인크리멘터.

### 【청구항 3】

제 1항에 있어서, 상기 소정의 논리 조합은,

논리식 ,



IF ZD = "0" and CA = "0", IO = "0000",

IF ZD = "1" and CA = "0", IO = IN+1,

IF CA = "1"(ZD = don't care), IO = IN

(여기서, ZD는 4비트 그룹별 제1 논리 상태 존재 정보, CA는 4비트 그룹별 플래그 정보, IO는 전체 인크리먼트 값의 4비트 그룹별 출력, IN은 오퍼랜드, "IN+1"은 인크리먼트 값, "0"은 제1 논리 상태, "1"은 제2 논리 상태)

에 의하여 수행되는 것을 특징으로 하는 인크리멘터.

#### 【청구항 4】

제 1항에 있어서, 상기 인크리멘터는,

상기 4비트 제로 검출부, 상기 플래그 정보 생성부, 상기 4비트 인크리먼트부, 및 상기 인크리먼트 출력부가 소정의 클럭에 의하여 액티브 될 때, 동작하는 것을 특징으로 하는 인크리멘터.

#### 【청구항 5】

제 4항에 있어서, 상기 소정의 클럭이 비활성화 상태일 때,

상기 4비트 제로 검출부, 상기 플래그 정보 생성부, 상기 4비트 인크리먼트부, 및 상기 인크리먼트 출력부 각각의 출력 버퍼들 앞단에서 소정의 전압으로 프리차징되는 것을 특징으로 하는 인크리멘터.

#### 【청구항 6】

제 5항에 있어서, 상기 출력 버퍼들 각각은,

상기 프리차징 전압을 인버팅하는 인버터와 상기 인버터 출력의 제어를 받아 상기 인버터의 입력단에 상기 소정의 전압을 공급하는 PMOSFET를 구비하는 것을 특징으로 하는 인크리멘터.

【청구항 7】

제 6항에 있어서, 상기 인버터 출력은,

상기 소정의 클럭이 활성화 상태이고, 상기 출력 버퍼들 각각의 앞단과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET들 모두가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 하는 인크리멘터.

【청구항 8】

인크리멘터에 의하여, 입력되는 오퍼랜드를 하위부터 4비트씩 그룹들을 지어 그룹들 각각에 제1 논리 상태가 존재하는지를 판단하여, 상기 제1 논리 상태가 존재하면 제2 논리 상태로 발생시키고, 상기 제1 논리 상태가 존재하지 하지 않으면 상기 제1 논리 상태로 발생시켜 4비트 그룹별 제1 논리 상태 존재 정보를 출력하는 단계;

상기 인크리멘터에 의하여, 상기 4비트 그룹별 제1 논리 상태 존재 정보의 하위에 서 최초로 나타나는 상기 제2 논리 상태에 대응하여, 상기 제2 논리 상태가 속한 그룹부터 전체 하위 그룹까지에 상기 제1 논리 상태로 발생시키고, 바로 위 상위 그룹부터 전체 상위 그룹까지에 상기 제2 논리 상태로 발생시켜 4비트 그룹별 플래그 정보를 출력하는 단계;

상기 인크리멘터에 의하여, 입력되는 상기 오퍼랜드를 받아 상기 4비트 그룹별 소정의 인크리먼트를 수행하는 단계; 및

상기 인크리멘터에 의하여, 상기 오퍼랜드, 상기 4비트 그룹별 제1 논리 상태 존재 정보, 상기 4비트 그룹별 플래그 정보, 및 상기 소정의 인크리먼트 값을 소정의 논리 조합하여 상기 4비트 그룹별로, 제1 논리 상태인 4비트, 상기 오퍼랜드 그대로의 4비트, 또는 상기 소정의 인크리먼트 값의 4비트를 출력하여 전체 인크리먼트 값을 발생시키는 단계를 구비하는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

#### 【청구항 9】

제 8항에 있어서, 상기 소정의 인크리먼트는,

논리식,

IF IN <0> and IN<1> = "1",

(IN+1)<3:2> = INC<3:2>,

(IN+1) <1:0> = "00"

IF IN<0> and IN<1> = "0",

(IN+1) <3:2> = IN<3:2> ,

(IN+1)<1:0> = INC<1:0>

( 여기서 IN은 오퍼랜드, "IN+1"은 인크리먼트 값, INC는 새로 정의된 인크리먼트, "0"은 제1 논리 상태, "1"은 제2 논리 상태)

및, 논리식,

INC <0> = ~IN<0>

INC<1> = IN<0> OR IN<1>

$INC <2> = \sim IN <2>$

$INC <3> = IN <2> OR IN <3>$

(여기서 IN은 오퍼랜드, INC는 새로 정의된 인크리먼트).

에 의하여 수행되는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

#### 【청구항 10】

제 8항에 있어서, 상기 소정의 논리 조합은,

논리식,

IF ZD = "0" and CA = "0", IO = "0000",

IF ZD = "1" and CA = "0", IO = IN+1,

IF CA = "1"(ZD = don't care), IO = IN

(여기서, ZD는 4비트 그룹별 제1 논리 상태 존재 정보, CA는 4비트 그룹별 플래그 정보, IO는 전체 인크리먼트 값의 4비트 그룹별 출력, IN은 오퍼랜드, "IN+1"은 인크리먼트 값, "0"은 제1 논리 상태, "1"은 제2 논리 상태)

에 의하여 수행되는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

#### 【청구항 11】

제 8항에 있어서, 상기 인크리멘터는,

소정의 클럭이 액티브 될 때, 동작하는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

**【청구항 12】**

제 11항에 있어서, 상기 소정의 클럭이 비활성화 상태일 때,

상기 인크리멘터의 출력 버퍼들 앞단에서 소정의 전압으로 프리차징되는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

**【청구항 13】**

제 12항에 있어서, 상기 출력 버퍼들 각각은,

상기 프리차징 전압을 인버팅하는 인버터와 상기 인버터 출력의 제어를 받아 상기 인버터의 입력단에 상기 소정의 전압을 공급하는 PMOSFET를 구비하는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

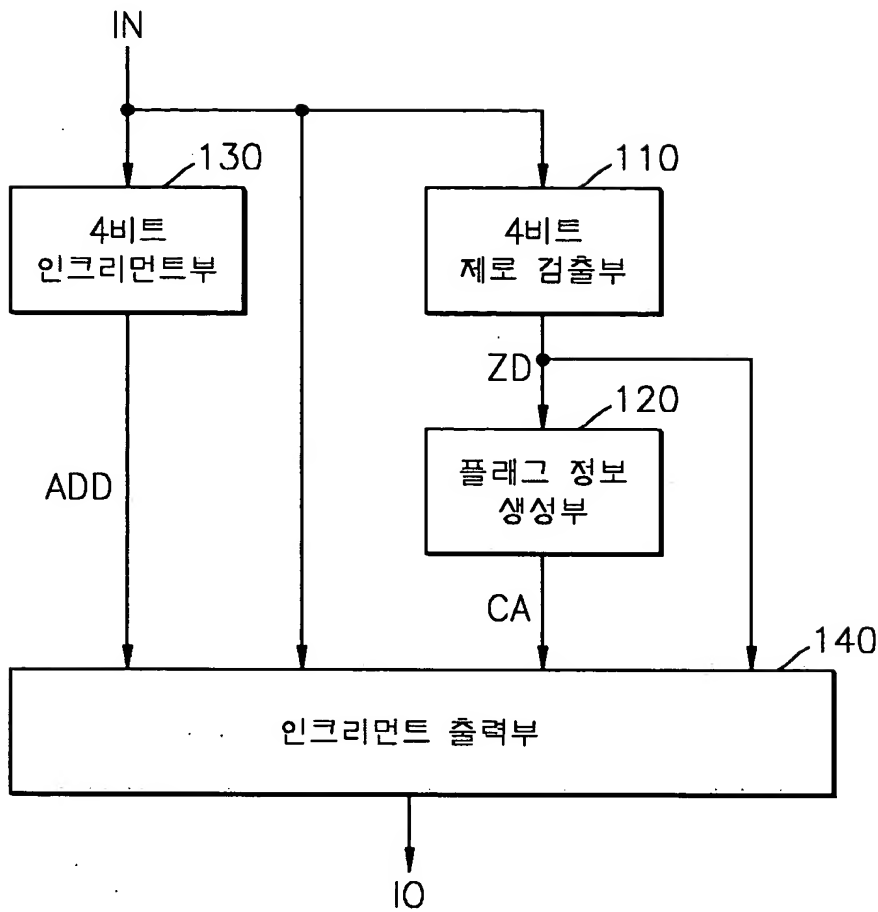
**【청구항 14】**

제 13항에 있어서, 상기 인버터 출력은,

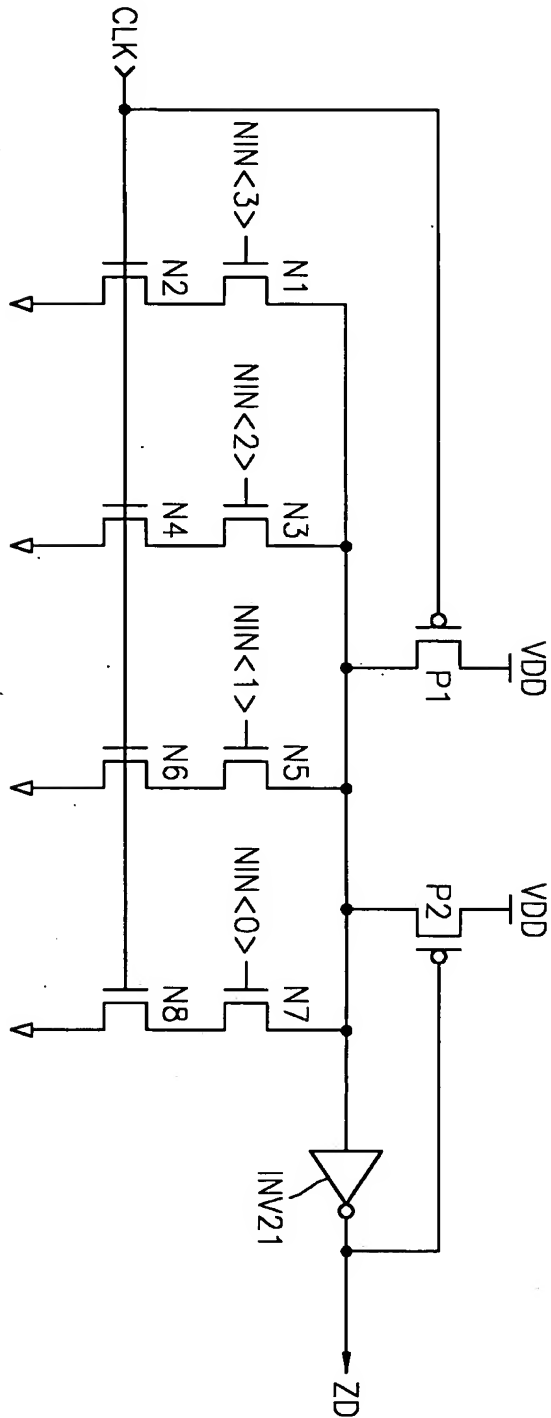
상기 소정의 클럭이 활성화 상태이고, 상기 출력 버퍼들 각각의 앞단과 접지 사이에서 직렬 연결되어 있는 다수개의 NMOSFET들 모두가 활성화 될 때, 상기 제2 논리 상태로 변환되는 것을 특징으로 하는 인크리멘터의 인크리먼트 방법.

【도면】

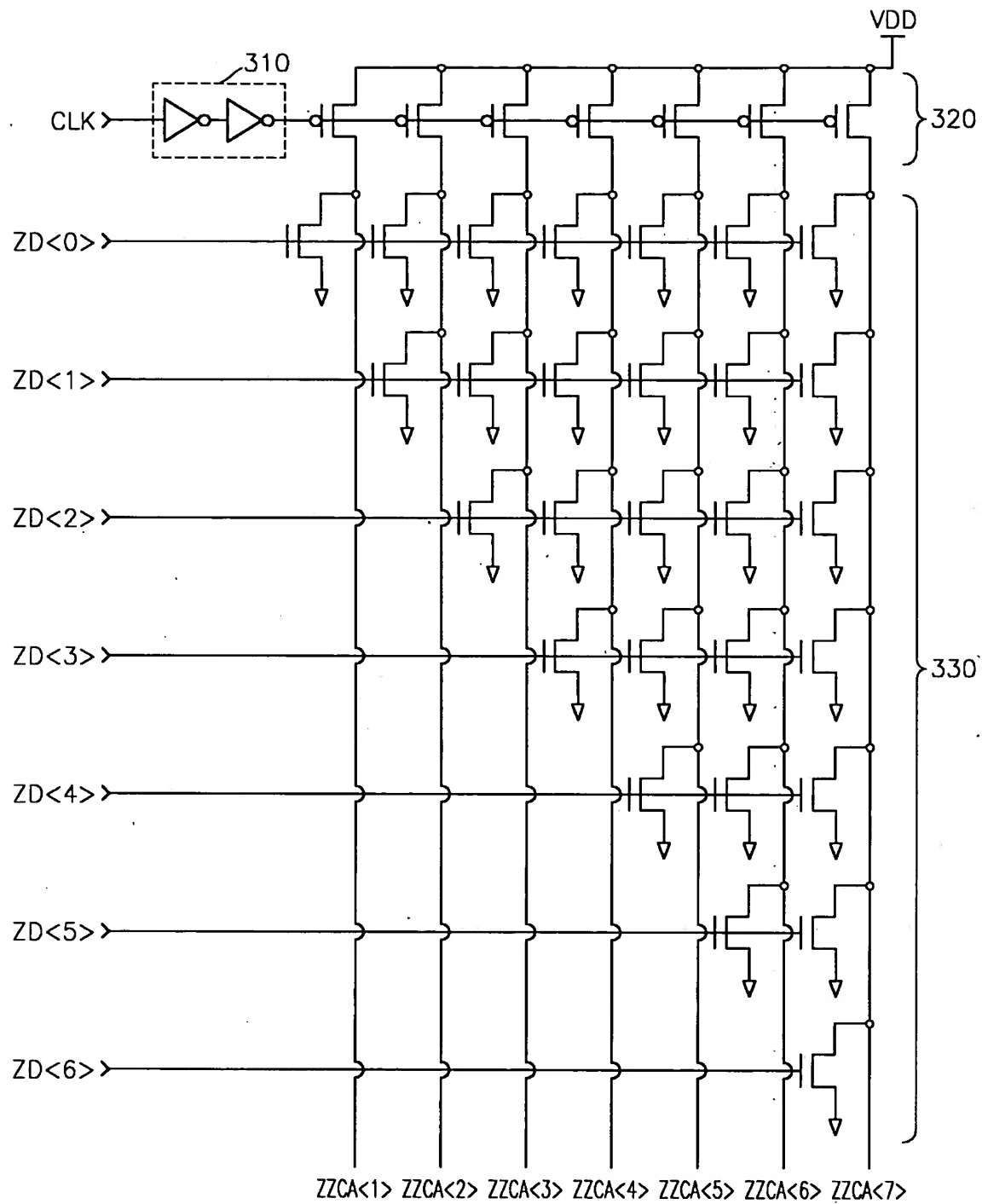
【도 1】



【도 2】

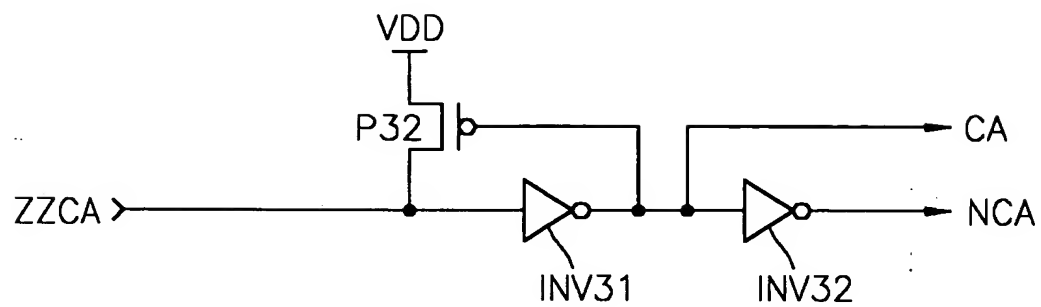


【도 3a】

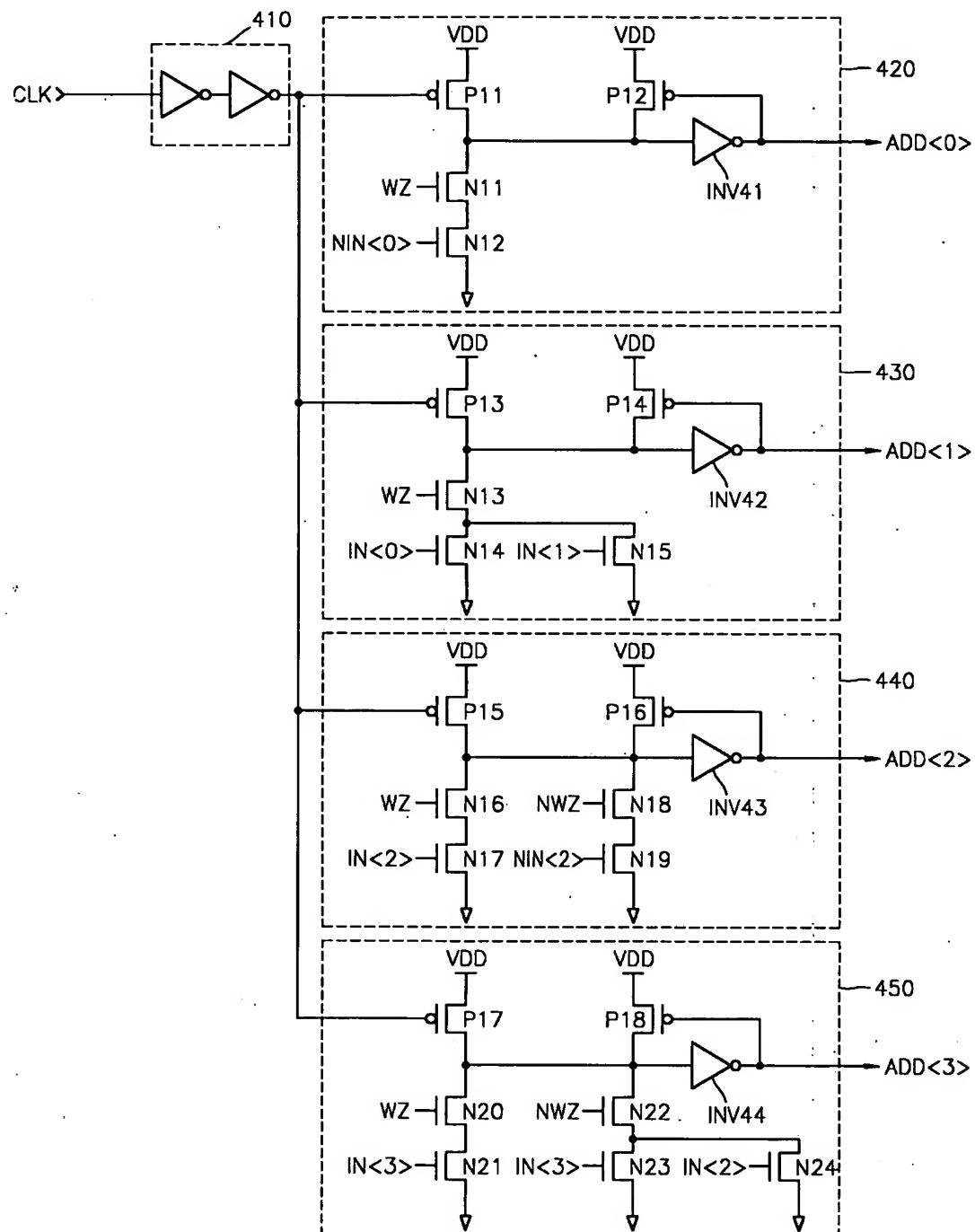




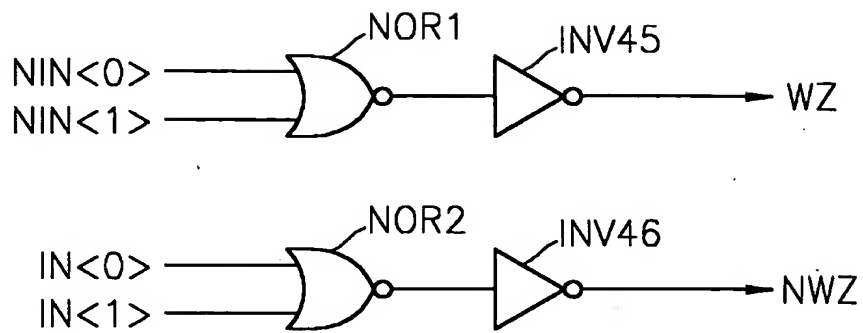
【도 3b】



【도 4a】



【도 4b】



【도 5】

